UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/845,750	04/30/2001	Walter Dixon III	345708004US	3459

28062	7590	06/28/2006

BUCKLEY, MASCHOFF, TALWALKAR LLC
5 ELM STREET
NEW CANAAN, CT 06840

EXAMINER
ZHEN, LI B

ART UNIT	PAPER NUMBER
2194	

DATE MAILED: 06/28/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

		Application No.	Applicant(s)
Office Action Summary		09/845,750	DIXON ET AL.
		Examiner	Art Unit	
		Li B. Zhen	2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>07 April 2006</u>.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *4-8,10-12,14-19,21,23-32,37-40 and 46-67* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *4-8,10-12,14-19,21,23-32,37-40 and 46-58* is/are rejected.

7)☒ Claim(s) *59-67* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *6/20/01* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☒ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

1.   Claims 4– 8, 10 – 12 and 14 – 19, 21, 23 – 32, 37 – 40, and 46 – 67 are pending in the current application.

2.   In view of the Appeal Brief filed on 04/07/2006, PROSECUTION IS HEREBY REOPENED. New grounds of rejection are set forth below.

To avoid abandonment of the application, appellant must exercise one of the following two options:

(1) file a reply under 37 CFR 1.111 (if this Office action is non-final) or a reply under 37 CFR 1.113 (if this Office action is final); or,

(2) initiate a new appeal by filing a notice of appeal under 37 CFR 41.31 followed by an appeal brief under 37 CFR 41.37. The previously paid notice of appeal fee and appeal brief fee can be applied to the new appeal. If, however, the appeal fees set forth in 37 CFR 41.20 have been increased since they were previously paid, then appellant must pay the difference between the increased fees and the amount previously paid.

Specification

3.   The applicant recites co-pending application by its title and docket information (p. 2, lines 20-23 and p. 12, lines 19-20). Please update the information by including U.S. application serial numbers or patent numbers.

4.   The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

Allowable Subject Matter

5.   Claims 59 – 67 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

6.   The following is an examiner's statement of reasons for allowance:

The prior art of record does not expressly teach or render obvious the invention as recited in dependent claims 59-67.

The prior art teaches a computer system for processing request messages [col. 16, lines 2 – 23 of Merrick], a plurality of sub-applications forming an application [col. 14, line 57 – col. 15, line 10 of Merrick], a sub-application having a match criteria indicating when the sub-application should process a request ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick], a service routine [col. 12, lines 12 – 30 of Merrick], the sub-applications using disparate logic models [col. 14, line 57 – col. 15, line 10 of Merrick], a context for the application that is shared by the sub-applications [col. 10, line 47 – col. 11, line 5 of De Borst], and the requests are HTTP requests with a URL and the match criteria is a regular expression relating to the URL [col. 17, lines 10 – 55 of Merrick]. However, the prior art does not teach HTTP requests with a URL and match criteria that invokes at least two sub-applications that are ordered and invoking the service routines of at least two sub-application in the order of the sub-applications.

In addition, the prior art of record does not provide a basis of evidence for asserting a motivation that one of ordinary skill level in the art at the time the invention was made would have integrated or modified the computer system for dispatching requests to perform services with HTTP requests with a URL and match criteria that invokes at least two sub-applications that are ordered and invoking the service routines of at least two sub-application in the order of the sub-applications as recited in the context of dependent claims 59 – 67 including the base claim and intervening claims.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Claim Rejections - 35 USC § 103

7.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set
> forth in section 102 of this title, if the differences between the subject matter sought to be patented and
> the prior art are such that the subject matter as a whole would have been obvious at the time the
> invention was made to a person having ordinary skill in the art to which said subject matter pertains.
> Patentability shall not be negatived by the manner in which the invention was made.

8.   This application currently names joint inventors. In considering patentability of

the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of

the various claims was commonly owned at the time any inventions covered therein

were made absent any evidence to the contrary. Applicant is advised of the obligation

under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was

not commonly owned at the time a later invention was made in order for the examiner to

consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g)

prior art under 35 U.S.C. 103(a).

9.   **Claims 4-8, 10-12, 14-19, 21, 23-32, 37-40 and 46-58 are rejected under 35**

U.S.C. 103(a) as being unpatentable over U.S. Patent No. 7,028,312 to Merrick et

al. [hereinafter Merrick] in view of U.S. Patent No. 6,173,327 to De Borst et al.

[hereinafter De Borst].

10.   As to claim 4, Merrick teaches the invention substantially as claimed including a

method in a computer system for dispatching requests [col. 8, lines 17 – 30] to perform

services to sub-applications [use XML RPC to invoke web services; col. 16, lines 2 – 23]

that use different logic models [allows an integration server to access many different

kinds of services; col. 14, line 57 – col. 15, line 10] the method comprising:

      receiving a request from a client computer to perform a service [to use XML RPC

to invoke a service, the client machine must identify the service that the request

message is intended to invoke; col. 17, lines 10 - 55]; and

for a plurality of sub-applications, determining whether the received request should be dispatched to the sub-application [If the request message names the target service, the server must first determine the message's document type so that it knows which data item contains the target service; col. 18, lines 28 – 40]; and

when it is determined that the request should be dispatched to the sub-application [the server machine interprets the message as a request to invoke a service], invoking a service routine of the sub-application passing the request [the server machine interprets the message as a request to invoke a service and invokes the service; col. 12, lines 12 - 30];

wherein the determining includes determining whether a match criteria ["interface Y" identifies the group of services…"DoSomething" identifies the name of the service] for the sub-application matches the received request ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 - 55];

wherein the requests are HTTP requests with a URL and the match criteria is a regular expression relating to the URL [URL illustrates how one may invoke a service named "DoSomething" on the webMethods B2B Integration Server: http://b2b.companyX.com/invoke/interfaceY/DoSomething; col. 17, lines 10 - 55].

Although Merrick teaches the invention substantially as claimed, Merrick does not teach providing a context for the sub-applications and the sub-applications share the provided context.

However, in the analogous art, De Borst teaches a distributed computing environment [col. 5, lines 28 – 63], client requesting a server to perform a service at a server [Object activation requests arise when an object call from a client or server application must be satisfied; col. 6, lines 8 – 25], providing a context for the sub-applications [Context structure to permit the run-time transfer of predefined information ("context") between components; col. 10, line 47 – col. 11, line 5], and the sub-applications share the provided context [context is passed from component to component using the context structure; col. 10, line 58 – col. 11, line 5].

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to modify the invention of Merrick to incorporate the features of providing a context for the sub-applications and the sub-applications share the provided context as taught by De Borst because this allows various components to deliver information in a protocol-independent manner across heterogeneous platforms and facilitates the creation of scalable and extensible components that can be adapted to numerous information delivery scenarios [col. 4, lines 3 - 11 of De Borst].

11.   As to claim 21, Merrick as modified teaches a computer system for dispatching HTTP [col. 8, lines 17 – 30 of Merrick] requests to sub-applications [use XML RPC to invoke web services; col. 16, lines 2 – 23 of Merrick], comprising:

a configuration file having a class [Configured objects have their implementation details stored in a repository (such as the MSF Warehouse 75) or in initialization scripts. Given a request for a specific object reference, the CEE 61 starts the appropriate capsule based on this configuration data; col. 5, line 62 – col. 6, line 10 of De Borst], initialization parameters [col. 11, lines 22 – 38 of De Borst], and a match criteria associated with the sub-applications ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick];

an initialization component that instantiates an object of the class for each sub-application in the configuration file [calls the information provider Factory to manufacture an information provider Stream object; col. 21, lines 36 – 60 of De Borst], the instantiated object being initialized with the initialization parameters for the sub-application [various components are initialized and loaded. In step 803, the implementation libraries are loaded and their initialization routines are called; col. 18, lines 25 – 40 of De Borst] and being provided with a context object [context parameter is an initialized context structure; col. 12, lines 15 – 27 of De Borst], the context object being shared by the instantiated objects so that the sub-applications share a common

context [Context structure to permit the run-time transfer of predefined information ("context") between components; col. 10, line 47 – col. 11, line 5 of De Borst]; and

a dispatcher that receives HTTP requests from client computers [to use XML RPC to invoke a service, the client machine must identify the service that the request message is intended to invoke; col. 17, lines 10 – 55 of Merrick] and, when the received HTTP request matches a match criteria of a sub-application ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick], invokes a service routine of the instantiated object of the class associated with the sub-application [the server machine interprets the message as a request to invoke a service and invokes the service; col. 12, lines 12 – 30 of Merrick];

wherein the match criteria is a regular expression relating to a URL of the HTTP request [URL illustrates how one may invoke a service named "DoSomething" on the webMethods B2B Integration Server: http://b2b.companyX.com/invoke/interfaceY/DoSomething; col. 17, lines 10 – 55 of Merrick]. As to the motivation for combining Merrick with De Borst, see the rejection to claim 4 above.

12.   As to claim 28, Merrick as modified teaches a computer system for processing request messages [use XML RPC to invoke web services; col. 16, lines 2 – 23 of Merrick], comprising:

a plurality of sub-applications forming an application [allows an integration server to access many different kinds of services; col. 14, line 57 – col. 15, line 10 of Merrick], a sub-application having a match criteria indicating when the sub-application should process a request ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick] and having a service routine to invoke when the match criteria indicates that the sub-application should process the request [the server machine

interprets the message as a request to invoke a service and invokes the service; col. 12, lines 12 – 30 of Merrick], the sub-applications using disparate logic models [allows an integration server to access many different kinds of services; col. 14, line 57 – col. 15, line 10 of Merrick];

a context for the application that is shared by the sub-applications [Context structure to permit the run-time transfer of predefined information ("context") between components; col. 10, line 47 – col. 11, line 5 of De Borst]; and

a dispatcher that receives requests from client computers [to use XML RPC to invoke a service, the client machine must identify the service that the request message is intended to invoke; col. 17, lines 10 – 55 of Merrick], evaluates the match criteria to identify which sub-applications have match criteria that match the requests ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick], and invokes the service routines of the identified sub-applications wherein invoked sub-applications use the context [the server machine interprets the message as a request to invoke a service and invokes the service; col. 12, lines 12 – 30 of Merrick];

wherein the requests are HTTP requests with a URL and the match criteria is a regular expression relating to the URL [URL illustrates how one may invoke a service named "DoSomething" on the webMethods B2B Integration Server: http://b2b.companyX.com/invoke/interfaceY/DoSomething; col. 17, lines 10 – 55 of Merrick]. As to the motivation for combining Merrick with De Borst, see the rejection to claim 4 above.

13.   As to claim 37, Merrick as modified teaches a computer system for processing request messages [use XML RPC to invoke web services; col. 16, lines 2 – 23 of Merrick], comprising:

a plurality of service means for servicing requests [allows an integration server to access many different kinds of services; col. 14, line 57 – col. 15, line 10 of Merrick], the

service means forming an application, each service means having a match criteria indicating when the service means should be invoked ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick], the service means implementing different logic models [allows an integration server to access many different kinds of services; col. 14, line 57 – col. 15, line 10 of Merrick]; and

dispatch means for receiving requests from client computers [to use XML RPC to invoke a service, the client machine must identify the service that the request message is intended to invoke; col. 17, lines 10 – 55 of Merrick], evaluating match criteria to identify which service means have match criteria that match the requests ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick], and invoking the identified service means whereby the service means share a context that is common to the service means of the application [the server machine interprets the message as a request to invoke a service and invokes the service; col. 12, lines 12 – 30 of Merrick];

wherein the requests are HTTP requests with a URL and the match criteria is a regular expression relating to the URL [URL illustrates how one may invoke a service named "DoSomething" on the webMethods B2B Integration Server: http://b2b.companyX.com/invoke/interfaceY/DoSomething; col. 17, lines 10 – 55 of Merrick]. As to the motivation for combining Merrick with De Borst, see the rejection to claim 4 above.

14.   As to claim 46, Merrick as modified teaches a computer-readable medium for controlling a computer system to dispatch requests to perform services to service routines [use XML RPC to invoke web services; col. 16, lines 2 – 23 of Merrick], by a method comprising:

receiving a request from a client computer to perform a service [to use XML RPC to invoke a service, the client machine must identify the service that the request message is intended to invoke; col. 17, lines 10 – 55 of Merrick]; and

for a plurality of service routines, retrieving a match criteria for the service routine [If the request message names the target service, the server must first determine the message's document type so that it knows which data item contains the target service; col. 18, lines 28 – 40 of Merrick];

determining whether the received request matches the retrieved match criteria ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick];

when it is determined that the request matches the retrieved match criteria ["interface Y" identifies the group of services..."DoSomething" identifies the name of the service of Merrick], invoking the service routine for processing of the received request [the server machine interprets the message as a request to invoke a service and invokes the service; col. 12, lines 12 – 30 of Merrick] whereby the service routines form an application [allows an integration server to access many different kinds of services; col. 14, line 57 – col. 15, line 10 of Merrick] and share a common context [Context structure to permit the run-time transfer of predefined information ("context") between components; col. 10, line 47 – col. 11, line 5 of De Borst];

wherein the requests are HTTP requests with a URL and the match criteria is a regular expression relating to the URL [URL illustrates how one may invoke a service named "DoSomething" on the webMethods B2B Integration Server: http://b2b.companyX.com/invoke/interfaceY/DoSomething; col. 17, lines 10 – 55 of Merrick]. As to the motivation for combining Merrick with De Borst, see the rejection to claim 4 above.

15.   As to claim 5, Merrick as modified teaches suppressing the invoking of additional service routines when an invoked service routine returns an indication to suppress the

invoking of additional service routines [col. 15, lines 45 – 55 and col. 20, lines 35 – 65 of
De Borst].

16.   As to claim 6, Merrick as modified teaches suppressing the invoking of additional
service routines when an invoked service routine responds to the received request [col.
15, lines 45 – 55 and col. 20, lines 35 – 65 of De Borst].

17.   As to claim 7, Merrick as modified teaches an invoked service routine performs
user authentication and indicates to suppress invoking of additional service routines
when a user cannot be authenticated [col. 15, lines 45 – 55 and col. 20, lines 35 – 65 of
De Borst].

18.   As to claim 8, Merrick as modified does not teach an invoked service routine logs
the received request. However, the logging of received requests is well known in the art
(access logs for a web server).
        It would have been obvious to one of ordinary skill in the art at the time the
invention was made to invoke service routine log received requests in the system of
Merrick as modified in order to decrease the costs associated with debugging the
system.

19.   As to claim 10, Merrick as modified teaches transforming the received request
from one protocol to another [col. 12, lines 28 – 48 of Merrick].

20.   As to claim 11, Merrick as modified teaches for each of a plurality of sub-
applications [use XML RPC to invoke web services; col. 16, lines 2 – 23 of Merrick],
retrieving initialization parameters for the sub-application [col. 11, lines 22 – 38 of De
Borst];
        retrieving an indication of a class for the sub-application ["invoke" informs the
server that the server is to invoke a service, "interfaceY" identifies the group of services

to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick]; and

instantiating an instance of the class with the retrieved initialization parameters [various components are initialized and loaded. In step 803, the implementation libraries are loaded and their initialization routines are called; col. 18, lines 25 – 40 of De Borst].

21.   As to claim 12, Merrick as modified teaches that the match criteria is in a configuration file for the sub-application [Configured objects have their implementation details stored in a repository (such as the MSF Warehouse 75) or in initialization scripts. Given a request for a specific object reference, the CEE 61 starts the appropriate capsule based on this configuration data; col. 5, line 62 – col. 6, line 10 of De Borst].

22.   As to claims 14, 15 and 16, Merrick as modified does not teach an interaction-based model, an action-view model or a workflow-based model. However, these logic models are all well known in the art.

It would have been obvious at the time of the invention to use these logic models in the sub-applications of Merrick as modified in order to use the architecture that is most appropriate for handling different requests.

23.   As to claim 17, Merrick as modified teaches the sub-applications form an overall application [allows an integration server to access many different kinds of services; col. 14, line 57 – col. 15, line 10 of Merrick] and the provided context is an application-level context [Context structure to permit the run-time transfer of predefined information ("context") between components; col. 10, line 47 – col. 11, line 5 of De Borst].

24.   As to claim 18, Merrick as modified teaches the sub-applications form an overall application that is web-based [use XML RPC to invoke web services; col. 16, lines 2 – 23 of Merrick].

25.   As to claim 19, Merrick as modified teaches the request is received from a web-server environment [col. 16, lines 22 – 49 of Merrick].

26.   As to claim 23, this is rejected for the same reasons as claims 5 and 11 above.

27.   As to claim 24, this is rejected for the same reasons as claims 6 and 11 above.

28.   As to claim 25, this is rejected for the same reasons as claims 11 and 14 above.

29.   As to claim 26, this is rejected for the same reasons as claim 11 and 15 above.

30.   As to claim 27, Merrick as modified teaches each of the sub-applications implement the same interface ["interfaceY" identifies the group of services to which the service belongs; col. 17, lines 10 - 55 of Merrick].

31.   As to claim 29, Merrick as modified teaches including an initialization component that instantiates an object of a specified class for each sub-application [calls the information provider Factory to manufacture an information provider Stream object; col. 21, lines 36 – 60 of De Borst].

32.   As to claim 30, Merrick as modified teaches the initialization component accesses configuration information [various components are initialized and loaded. In step 803, the implementation libraries are loaded and their initialization routines are called; col. 18, lines 25 – 40 of De Borst] that specifies the class of each sub-application and any initialization parameters for the sub-applications [calls the information provider Factory to manufacture an information provider Stream object; col. 21, lines 36 – 60 of De Borst].

33.   As to claim 31, Merrick as modified teaches a context object representing the context and wherein the initialization component provides the context object to each

sub-application [context is passed from component to component using the context structure; col. 10, line 58 – col. 11, line 5 of De Borst].

34.   As to claim 32, Merrick as modified teaches each service routine is passed a request parameter [input arguments to the service were passed as CGI query parameters to a target web site; col. 11, line 57 – col. 12, line 15 of Merrick] and returns a response parameter [service being invoked returns output arguments; col. 14, lines 20 – 40 of Merrick].

35.   As to claims 38 – 40, these are rejected for the same reasons as claims 29, 30 and 32 respectively, see the rejections to claims 29, 30 and 32 above.

36.   As to claims 47 – 48, these are product claims that correspond to method claims 5 – 6; note the rejections to claims 5 – 6 above, which also meet these product claims.

37.   As to claims 49 – 53, Merrick as modified teaches all of the sub-applications or service means or service routines execute on the same server computer ["invoke" informs the server that the server is to invoke a service, "interfaceY" identifies the group of services to which the service belongs, and "DoSomething" identifies the name of the service to invoke from this group; col. 17, lines 9 – 55 of Merrick].

38.   As to claims 54 – 58, Merrick as modified teaches a respective service routine is invoked for the request with respect to each of at least two of the sub-applications [use XML RPC to invoke web services; col. 16, lines 2 – 23 and col. 17, lines 9 – 55 of Merrick].

Conclusion

39.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Patent No. 6,012,090 discloses registration applet embedded in a registration page of a browser program that allows a user to associate a user-specified group name with a plurality of UTRLs, HTTP POST or GET requests or other network service identifiers, such that the group name designates a category of information provided by corresponding network services.
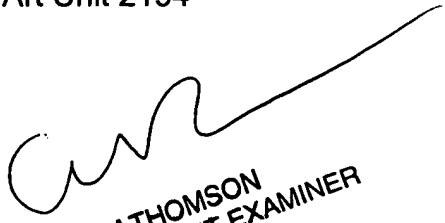
CONTACT INFORMATION

40.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (571) 272-3768. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Thomson can be reached on 571-272-3718. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Li B. Zhen
Examiner
Art Unit 2194

lbz

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER